

# INSTITUT FÜR INFORMATIK

## **POD-DEIM-Verfahren zur Modellreduktion und Anwendung auf Metos3D**

Peters N., Piwonski J., Slawig T.

Bericht Nr. 1603

April 2016

ISSN 2192-6247



CHRISTIAN-ALBRECHTS-UNIVERSITÄT  
ZU KIEL

Institut für Informatik der  
Christian-Albrechts-Universität zu Kiel  
Olshausenstr. 40  
D – 24098 Kiel

## **POD-DEIM-Verfahren zur Modellreduktion und Anwendung auf Metos3D**

Peters N., Piwonski J., Slawig T.

Bericht Nr. 1603  
April 2016  
ISSN 2192-6247

e-mail: {npj,jpi,ts}@informatik.uni-kiel.de

Dieser Bericht enthält die schriftliche Ausarbeitung  
zum Masterprojekt von Neel Peters.

CHRISTIAN-ALBRECHTS-UNIVERSITÄT ZU  
KIEL

MASTERPROJEKT

NUMERISCHE MATHEMATIK UND OPTIMIERUNG

---

POD-DEIM-Verfahren zur  
Modellreduktion und Anwendung auf Metos3d

---

*Author:*

Neel PETERS

*Supervisor:*

Prof. Thomas SLAWIG

17. April 2016

# Inhaltsverzeichnis

<b>1</b>	<b>Problemdefinition</b>	<b>2</b>
<b>2</b>	<b>Reduzierte Basis Methoden</b>	<b>2</b>
<b>3</b>	<b>Proper Orthogonal Decomposition (POD)</b>	<b>3</b>
<b>4</b>	<b>Discrete Empirical Interpolation Method (DEIM)</b>	<b>5</b>
<b>5</b>	<b>Experimente mit Metos3d</b>	<b>7</b>
5.1	Experiment 1 . . . . .	7
5.2	Experiment 2 . . . . .	9
5.3	Experiment 3 . . . . .	12
<b>6</b>	<b>Komplexität und Konvergenz</b>	<b>14</b>
<b>7</b>	<b>Code</b>	<b>16</b>
<b>8</b>	<b>Fazit</b>	<b>16</b>

# 1 Problemdefinition

Gegeben sei ein System von nicht-linearen Differenzialgleichungen der Form

$$\frac{d}{dt}y(t) = Ay(t) + f(y(t)), \quad (1)$$

dabei ist  $t \in [0, T]$ ,  $y : \mathbb{R} \rightarrow \mathbb{R}^n$ ,  $y(0) = y_0 \in \mathbb{R}^n$ ,  $A \in \mathbb{R}^{n \times n}$ , die diskrete Approximation des linearen Differenzialoperators,  $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$  eine nicht lineare Funktion. Um die Modelldimension zu reduzieren, wird mit einer Galerkin-Projektion das System auf einen niedrig dimensionalen Unterraum projiziert. Für ein  $k \ll n$  sei  $V \in \mathbb{R}^{n \times k}$  eine Matrix orthonormaler Vektoren, deren Spalten die Basisvektoren der reduzierten Basis sind. Durch das Ersetzen von  $y(t)$  in 1 durch  $V\hat{y}(t)$  mit  $\hat{y}(t) \in \mathbb{R}^k$  wird das System auf  $V$  projiziert. Aus (1) wird dann

$$\frac{d}{dt}\hat{y}(t) = V^T AV\hat{y}(t) + V^T f(V\hat{y}(t)). \quad (2)$$

Für den linearen Anteil, also  $V^T AV = \hat{A} \in \mathbb{R}^{k \times k}$ , ist eine Reduzierung der Komplexität damit schon erreicht. Der nichtlineare Anteil  $V^T f(V\hat{y}(t))$  muss immer noch hochdimensional ausgewertet werden, da  $V\hat{y} \in \mathbb{R}^n$  liegt.

## 2 Reduzierte Basis Methoden

Um eine reduzierte Basis für ein Problem zu erstellen, wird zunächst ein Satz von Lösungen des hochdimensionalen Systems erstellt, sogenannte Snapshots.

$$S = \{y_1, \dots, y_m\} \quad (3)$$

Zu dem Raum, den  $S$  aufspannt, wird eine Basis  $\{v_1, \dots, v_r\}$  erzeugt, so das gilt:

$$V_r = \text{span}\{v_1, \dots, v_r\} = \text{span}\{y_1, \dots, y_m\}. \quad (4)$$

Mit Hilfe dieser Basis und den Koeffizienten  $c_1, \dots, c_n \in \mathbb{R}$  kann nun eine reduzierte Lösung in  $V_r$  bestimmt werden durch Linearkombination der Basisvektoren

$$\hat{y} = \sum_{i=1}^r c_i v_i = Vc \quad \text{mit} \quad V = \{v_1, \dots, v_r\}, c = [c_1, \dots, c_r]^T. \quad (5)$$

Eine Möglichkeit, eine Basis von einem Satz Snapshots zu erstellen, ist das POD-Verfahren. Die Koeffizienten können durch eine Galerkin-Projektion bestimmt werden.

### 3 Proper Orthogonal Decomposition (POD)

Sei  $\{y_1, \dots, y_m\} \subset \mathbb{R}^n$  eine Reihe von Snapshots und  $Y = [y_1 | \dots | y_m] \in \mathbb{R}^{n \times m}$  die zugehörige Matrix. Das POD-Verfahren liefert nun eine Orthonormalbasis für den Span der Snapshots. Sei  $r = \text{rang}\{Y\}$ , dann existieren mit  $k \leq r$ ,  $\{v_i\}_i^k \subset \mathbb{R}^n$  orthonormale Basisvektoren.

**Definition 1. (SVD):** Sei  $A \in \mathbb{R}^{m \times n}$  eine reelle Matrix Rang  $p$ , dann existieren zwei orthogonal Matrizen

$$U = [\varphi_1 | \dots | \varphi_n] \in \mathbb{R}^{m \times m}, \quad Z = [\Psi_1 | \dots | \Psi_m] \in \mathbb{R}^{n \times n}$$

so das

$$A = U \Sigma Z^T, \text{ mit } \Sigma = \text{diag}(\sigma_1, \dots, \sigma_p) \in \mathbb{R}^{m \times n} \quad (6)$$

und es gilt  $\sigma_1 \geq \dots \geq \sigma_p > 0$  und auch  $A \Psi_i = \sigma_i \varphi_i$  und  $A^T \varphi_i = \sigma_i \Psi_i$  mit  $i = 1, \dots, p$ .

Die Singulärwertzerlegung (SVD) von  $Y = U \Sigma Z^T$  mit den orthogonal Matrizen  $U = [\varphi_1 | \dots | \varphi_n] \in \mathbb{R}^{n \times n}$  und  $Z = [\Psi_1 | \dots | \Psi_m] \in \mathbb{R}^{m \times m}$  und der Diagonalmatrix  $\Sigma = \text{diag}(\sigma_1 \geq \dots \geq \sigma_r)$  mit den Singulärwerten  $\sigma_1 \geq \dots \geq \sigma_r$  liefert diese Basisvektoren. Für ein  $k \leq m$  ist die POD-Basis definiert als Menge der ersten  $k$  linken Singulärvektoren  $\varphi_1, \dots, \varphi_k$  von  $U$ . Sei  $V = [v_1, \dots, v_k] \in \mathbb{R}^{n \times k}$  die aus den POD-Basisvektoren bestehende Matrix, eine Approximation eines Snapshots  $y_j$  im  $\text{span}\{Y\}$  ist dann  $Vc$  mit Koeffizienten  $c \in \mathbb{R}^k$ . Die Galerkin Orthogonalität des Restterms  $y_j - Vc$  liefert  $V^T(y_j - Vc) = 0$ , damit folgt die Approximation  $y_j \approx VV^T y_j$ . Das POD-Verfahren liefert eine optimale orthonormale Basis in Hinblick auf die Summe des quadratischen Fehlers der Approximation von den Snapshots.

**Satz 1. (POD-Fehler):** Sei  $\mathbb{V}_k = \{W \in \mathbb{R}^{n \times k} : W^T W = I_k\}$  die Menge aller  $k$ -dimensionalen Orthonormalbasen und  $V \in \mathbb{R}^{n \times k}$  die Matrix von Basisvektoren, die das POD-Verfahren zu den Snapshots  $Y = \{y_1, \dots, y_m\}$  liefert, dann gilt,

$$\sum_{j=1}^m \|y_j - VV^T y_j\|_2^2 = \min_{W \in \mathbb{V}_k} \sum_{j=1}^m \|y_j - WW^T y_j\|_2^2 = \sum_{i=k+1}^p \sigma_i^2. \quad (7)$$

**Theorem 1. (Schmidt-Eckart-Young)** Sei  $A \in \mathbb{R}^{m \times n}$  eine Matrix mit Rang  $p$ , dann existieren die orthogonal Matrizen  $U = [\varphi_1 | \dots | \varphi_n] \in \mathbb{R}^{n \times n}$  und  $Z =$

$[\Psi_1 | \dots | \Psi_m] \in \mathbb{R}^{m \times m}$  und man kann  $A = U\Sigma Z^T$  schreiben mit der Diagonalmatrix  $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_p)$  und mit den Singulärwerten  $\sigma_1 \geq \dots \geq \sigma_p \geq 0$ . Die Matrix

$$A_k = \sum_{i=1}^k \sigma_i \varphi_i \Psi_i^T, \quad 0 \leq k \leq p \quad (8)$$

erfüllt die Eigenschaft

$$\|A - A_k\|_F = \min_{\substack{B \in \mathbb{R}^{m \times n} \\ \text{rank}(B) \leq k}} \|A - B\|_F = \sqrt{\sum_{i=k+1}^p \sigma_i^2}. \quad (9)$$

**Beweis von Satz 1:** Aus Theorem 1 folgt, dass die beste Rang  $k$  Approximation von  $Y$  gegeben ist durch

$$Y_k = \sum_{i=1}^k \sigma_i \varphi_i \Psi_i^T.$$

Mit  $\Psi_i = \frac{1}{\sigma_i} S^T \varphi_i$  folgt

$$Y_k = \sum_{i=1}^k \sigma_i \varphi_i \left( \frac{1}{\sigma_i} Y^T \varphi_i \right)^T = \sum_{i=1}^k \varphi_i \varphi_i^T Y = V V^T Y.$$

Außerdem gilt für ein  $W \in V_k$

$$\sum_{j=1}^m \|y_j - W W^T y_j\|_2^2 = \|Y - W W^T Y\|_F^2$$

und mit (9) gilt

$$\|Y - Y_k\|_F^2 = \min_{\substack{B \in \mathbb{R}^{m \times n} \\ \text{rank}(B) \leq k}} \|Y - B\|_F^2 \leq \min_{W \in \mathbb{V}_k} \|Y - W W^T Y\|_F^2.$$

Da  $Y_N = V V^T Y$  folgt

$$\|Y - V V^T Y\|_F^2 = \min_{W \in \mathbb{V}_k} \|Y - W W^T Y\|_F^2 = \sum_{i=k+1}^p \sigma_i^2.$$

Die Energie oder relative Information der ersten  $k$  POD-Basisvektoren kann beschrieben werden durch

$$I(k) = \frac{\sum_{i=1}^k \sigma_i^2}{\sum_{i=1}^p \sigma_i^2}. \quad (10)$$

## 4 Discrete Empirical Interpolation Method (DEIM)

**Definition 2.** Sei  $f : D \mapsto \mathbb{R}^n$  eine nicht-lineare vektorwertige Funktion mit  $D \subset \mathbb{R}^d$  für ein  $d \in \mathbb{N}$ . Sei  $\{u_l\}_{l=1}^m \subset \mathbb{R}^n$  ein Menge linear unabhängiger Vektoren. Für ein  $x \in D$  ist dann die DEIM Approximation der Ordnung  $m$  im Span von  $\{u_l\}_{l=1}^m$  gegeben durch:

$$\hat{f}(x) := U(P^T U)^{-1} P^T f(x) \quad (11)$$

mit  $U = [u_1 | \dots | u_m] \in \mathbb{R}^{n \times m}$  der Basis, die mit dem POD-Verfahren aus den nicht-linearen Snapshots erstellt wurde, und  $P = [e_{p_1} | \dots | e_{p_m}] \in \mathbb{R}^{n \times m}$ , wobei  $\{p_1, \dots, p_m\}$  der Output des DEIM-Algorithmus beim Input von  $\{u_l\}_{l=1}^m$  ist.

---

### Algorithm 1 DEIM

---

```

1: Input:  $\{u_i\}_{i=1}^m \subset \mathbb{R}^n$ 
2: Output:  $\vec{\varrho} = \{\varrho_1, \dots, \varrho_m\}^T \in \mathbb{N}^m$ 
3:  $[|p|\varrho_1] = \max\{|u_1|\}$ 
4:  $U = [u_1], P = [e_{\varrho_1}], \vec{\varrho} = [\varrho_1]$ 
5: for  $i = 2$  to  $m$  do
6:   Solve  $(P^T U)c = P^T u_i$  nach  $c$ 
7:    $r = u_i - Uc$ 
8:    $[|p|\varrho_i] = \max\{|r|\}$ 
9:    $U \leftarrow [U|u_i], P \leftarrow [P|e_{\varrho_i}], \vec{\varrho} \leftarrow \begin{bmatrix} \varrho \\ \varrho_i \end{bmatrix}$ 
10: end for

```

---

Die Notation der max-Funktion ist hier wie in Matlab, also bedeutet  $[|p|\varrho_l] = \max\{|u|\} \mid p = |r_{\varrho_l}| = \max_{i=1, \dots, n} \{|r_1 i|\}$ . Der Algorithmus 1 startet mit der Wahl des ersten Interpolationsindizes  $\varrho_1$ , der mit dem Index des größten Eintrag vom ersten Basisvektors  $u_1$  übereinstimmt. Die weiteren Indizes  $p_l$  für  $l = 2, \dots, m$  werden so gewählt, dass sie mit dem Index des größten Eintrag des Resttermes  $r = u_l - Uc$  korrelieren. Dass  $\hat{f}(x)$  in 11 eine Interpolation von der originalen Funktion  $f(x)$  ist, kann man leicht nachrechnen:

$$P^T \hat{f}(x) = P^T (U(P^T U)^{-1} P^T f(x)) = (P^T U)(P^T U)^{-1} P^T f(x) = P^T f(x).$$



Eine weitere Eigenschaft von der DEIM Interpolation ist, dass die Indizes, die durch  $\{u_i\}_{i=1}^m$  erzeugt wurden,  $\{\varrho_l\}_{l=1}^m$  unabhängig von der Basis des Raumes  $\text{span}\{u_1, \dots, u_m\}$  sind. Dafür sei  $\{q_l\}_{l=1}^m$  eine Basis von dem  $\text{span}\{u_1, \dots, u_m\}$ , mit  $Q = [q_1 | \dots | q_m]$  gilt dann

$$U(P^T U)^{-1} P^T f(x) = (Q(P^T Q)^{-1} P^T f(x)). \quad (12)$$

Da  $\langle Q \rangle = \langle U \rangle$  folgt  $U = QR$  mit einer nicht-singulären Matrix  $R \in \mathbb{R}^{m \times m}$  und damit auch

$$U(P^T U)^{-1} P^T f(x) = (QR((P^T Q)R)^{-1} P^T f(x)) = (Q(P^T Q)^{-1} P^T f(x)). \quad (13)$$

**Satz 2. (DEIM-Fehler):** Sei  $f \in \mathbb{R}^n$  ein Vektor, sei  $\{u_l\}_{l=1}^m \subset \mathbb{R}^n$  eine Menge linear unabhängiger orthogonaler Vektoren. Die DEIM Approximation der Ordnung  $m \leq n$  von  $f$  in dem Raum, der von  $\{u_l\}_{l=1}^m$  aufgespannt wird, ist  $\hat{f} = \mathbb{P}f$  mit  $\mathbb{P} = U(P^T U)^{-1} P^T$ ,  $U = [u_1 | \dots | u_m] \in \mathbb{R}^{n \times m}$  und  $P = [e_{p_1} | \dots | e_{p_m}] \in \mathbb{R}^{n \times m}$ , wobei  $\{p_1, \dots, p_m\}$  der Output des DEIM-Algorithmus beim Input von  $\{u_l\}_{l=1}^m$  ist. Dann gilt

$$\|f - \hat{f}\|_2 \leq C_m \varepsilon(f) \quad (14)$$

mit  $C_m = \|U(P^T U)^{-1} P^T\|_2$  und  $\varepsilon(f) = \|f - UU^T f\|_2$ .

**Beweis von Satz 2:** Um den Fehler in der 2-Norm zu beschränken, nehmen wir uns die beste Approximation von  $f$  im Spann von  $U$ , also  $f_* = UU^T f$ . Nun gilt

$$f = (f - f_*) + f_* = w + f_* \quad \text{mit} \quad w = f - f_* = f - UU^T f \quad (15)$$

und dann auch

$$\hat{f} = \mathbb{P}f = \mathbb{P}(w + f_*) = \mathbb{P}w + f_*, \quad (16)$$

da  $\mathbb{P}f_* = f_*$  ist. Damit folgt

$$\|f - \hat{f}\|_2 = \|w + f_* - (\mathbb{P}w + f_*)\|_2 \leq \|I - \mathbb{P}\|_2 \|w\|_2. \quad (17)$$

Es gilt damit

$$\|f - \hat{f}\|_2 \leq \|\mathbb{P}\|_2 \|w\|_2 = C_m \varepsilon(f). \quad (18)$$

Man kann  $C_m$  beschränken durch

$$C_m \leq \frac{(1 + \sqrt{2n})^{m-1}}{\|u_1\|_\infty}. \quad (19)$$

## 5 Experimente mit Metos3d

Für erste Untersuchungen wurde nur mit dem linearen Anteil von Metos3d und dem POD-Verfahren gearbeitet. Es wurden hierfür mit dem Simulation Package hochdimensionale Lösungen mit dem NULL Modell, das nur den linearen Anteil auswertet, ausgerechnet. Das Ganze wurde für verschiedene zufällige Anfangsverteilungen gemacht, die von Markus Pfeil erstellt wurden. Aus diesen Lösungen werden Snapshotsmatrizen erstellt und für diese mit dem POD-Verfahren die Basis für den Raum, den diese aufspannen, bestimmt. Dann werden die 12 Monatsmittel-Matrizen des reduzierten Systems vorberechnet und mit einem der zufälligen Anfangsverteilungen werden mehrere Zeitschritte und Spinups berechnet und diese mit den hochdimensionalen Lösungen verglichen. Für den Vergleich wurde die euklidische Norm verwendet und der Abstand der hochdimensionalen Lösung zu der reduzierten bestimmt, außerdem wurde der Abstand von einem Jahr zum Vorjahr bestimmt, jeweils für die hochdimensionalen und die reduzierten Trajektorien. Dies wurde gemacht um einzuschätzen, wie nah das Verhalten des reduzierten Modells an dem des hochdimensionalen Modells ist. Um das Verhalten des reduzierten Modells zu messen, wurde auch der Abstand zur optimalen Lösung berechnet. Als optimale Lösung wurde dabei eine Gleichverteilung angenommen, da dies der Zustand ist, gegen den das hochdimensionale Modell konvergiert.

### 5.1 Experiment 1

Es wurde eine Snapshotmatrix aus den Lösungen von 13 Simulationen mit verschiedenen Anfangsverteilungen mit je 5 Spinups erstellt. Dafür wurden die Zwischenlösungen am Ende jeden Monats genommen. Also insgesamt  $13 * 5 * 12 = 780$  Snapshots. Die SVD lieferte Singulärwerte, die in Abbildung 1 zu sehen sind. Diese fallen deutlich ab dem 607 Singulärwert ab, also wurden die ersten 607 linken Singulärvektoren als POD-Basis ausgewählt. Bestimmt wurde das durch

$$I(k) = \frac{\sum_{i=1}^k \sigma_i^2}{\sum_{i=1}^p \sigma_i^2} \geq 1 - \epsilon_{pod}^2, \quad (20)$$

wobei als Fehler  $\epsilon_{pod} = 10^{-3}$  gewählt wurde.

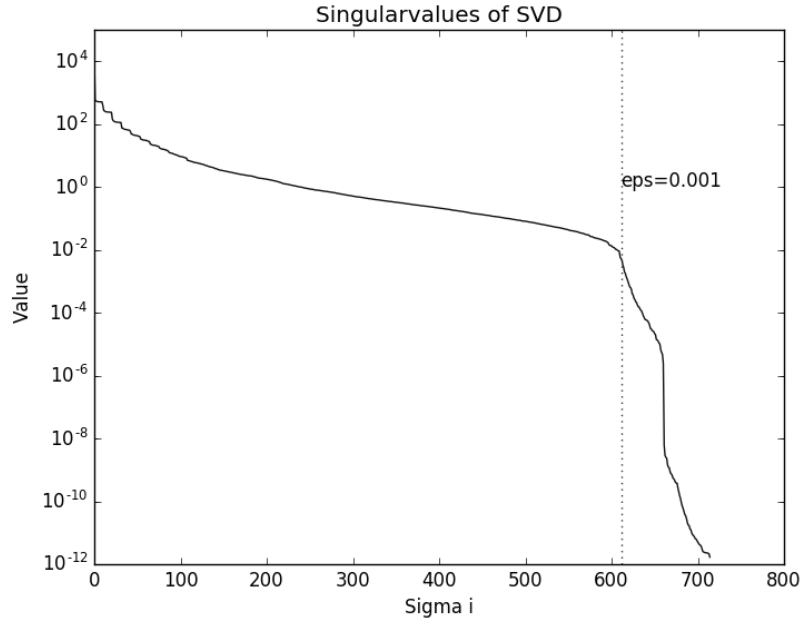


Abbildung 1: Singulärwerte von UD 25 5 12 für Experiment 1

Mit dieser Basis  $V$  wurden für eine Anfangsverteilung 78 Modelljahre gerechnet und die Lösung in jedem Jahr mit der hochdimensionalen verglichen, das heißt es wurde  $\|y_r - y_{hd}\|_2$  bestimmt. Außerdem wurde auch der Fehler zur Gleichverteilung und die Norm der Differenzen eines Jahres zum Vorjahr bestimmt. In Abbildung 2 sieht man deutlich, dass die Simulationen sich am Anfang ähnlich verhalten, dann aber ab 25 Jahren Modellzeit von einander abweichen. An diesem Punkt ist die maximale Genauigkeit erreicht, die mit der Basis möglich ist, dies kommt wahrscheinlich daher, dass beim Erstellen der Basis nur Lösungen von maximal 5 Modelljahren genommen wurden.

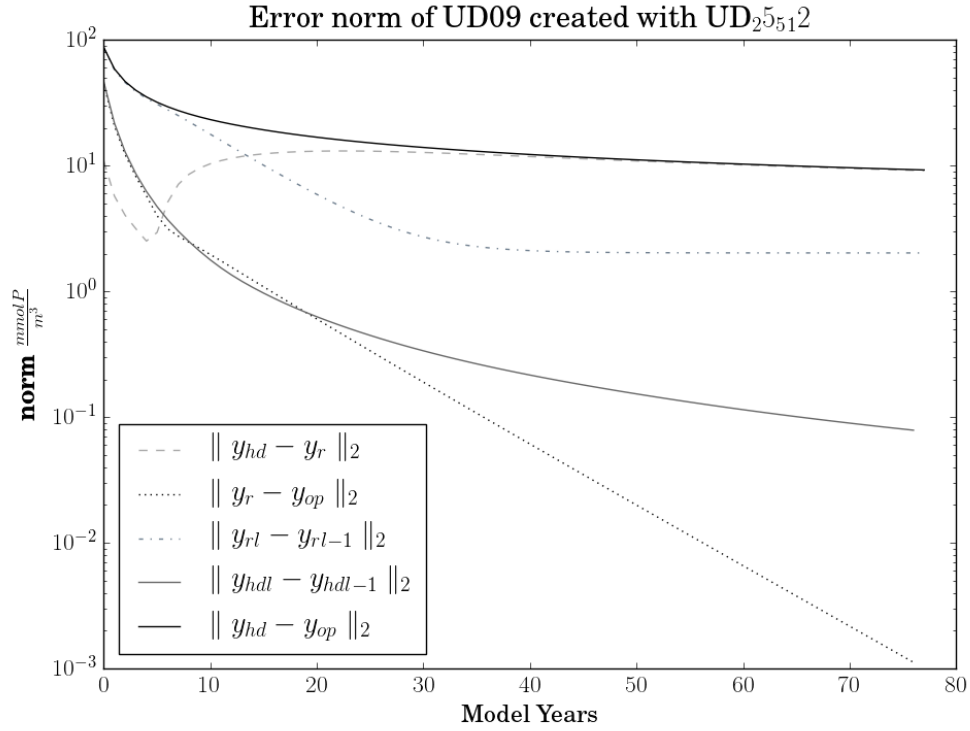


Abbildung 2: Vergleich des reduzierten Modelllaufs mit dem hochdimensionalen, Modellzeit 78 Jahre, gestrichelte Linie: Unterschied zwischen der hochdimensionalen und der reduzierten Lösung, graue gepunktete Linie: Fehler zur optimalen Verteilung des reduzierten Modelllaufs, gestrichelte und gepunktete Linie: Abstand zum Vorjahr, des reduzierten Modelllaufs, durchgezogene, graue Linie: Abstand zum Vorjahr des hochdimensionalen Modelllaufs, durchgezogene, schwarze Linie: Fehler zur optimalen Verteilung des hochdimensionalen Modelllaufs.

## 5.2 Experiment 2

Hier wurden Lösungen von einer längeren Simulation genommen. Es wurde mit einer Anfangsverteilung 78 Modelljahre gerechnet und wieder für jeden Monat eine Lösung zur Snapshotmatrix hinzugefügt, also  $78 * 12 = 936$  Snapshots.

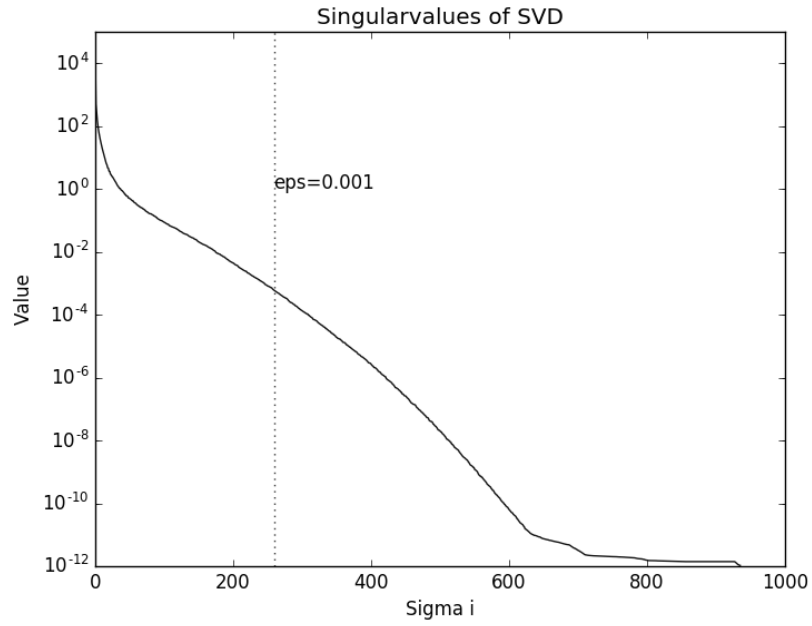


Abbildung 3: Singulärwerte von UD 09 78 12

Hier ist eine sehr deutliche Reduzierung möglich, bei einem Fehler  $\epsilon_{pod} = 1e-03$  kann die Basis von 936 auf 259 reduziert werden. Der Vergleich (Abbildung 4) von der reduzierten Lösung mit der hochdimensionalen zeigt, dass hier, aufgrund der längeren Trajektorie, bessere Ergebnisse erzielt werden. Allerdings gibt es Anfangsverteilungen, für die diese Basis Lösungen liefert, die stark von der hochdimensionalen abweichen (Abbildung 5).

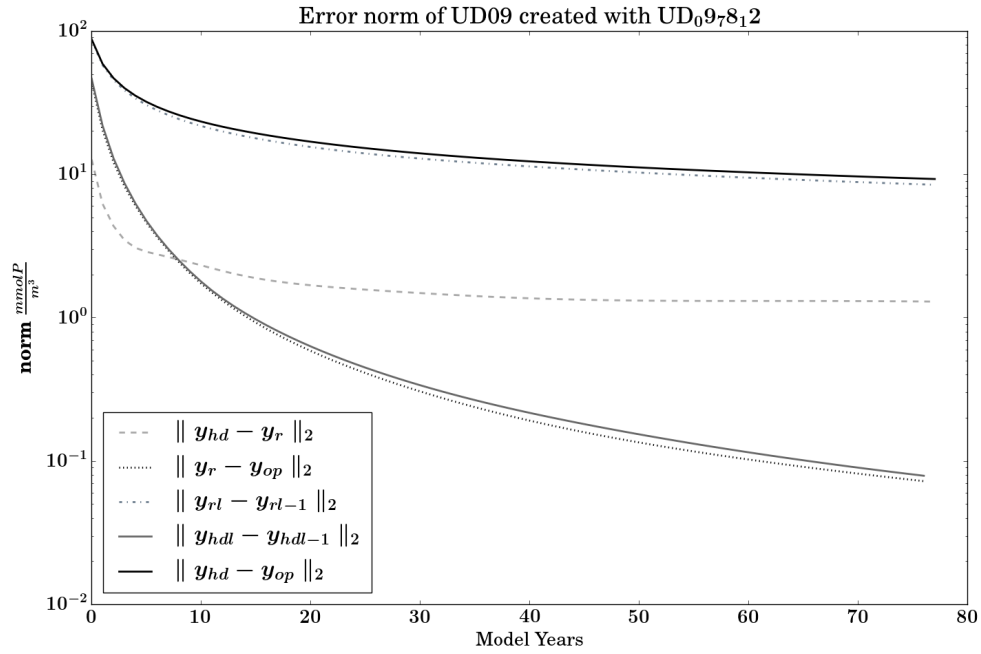


Abbildung 4: Vergleich des reduzierten Modelllaufs mit dem hochdimensionalen mit der Anfangsverteilung UD 09. Modellzeit 78 Jahre, gestrichelte Linie: Unterschied zwischen der hochdimensionalen und der reduzierten Lösung, graue gepunktete Linie: Fehler zur optimalen Verteilung des reduzierten Modelllaufs, gestrichelte und gepunktete Linie: Abstand zum Vorjahr, des reduzierten Modelllaufs, durchgezogene, graue Linie: Abstand zum Vorjahr des hochdimensionalen Modelllaufs, durchgezogene, schwarze Linie: Fehler zur optimalen Verteilung des hochdimensionalen Modelllaufs.

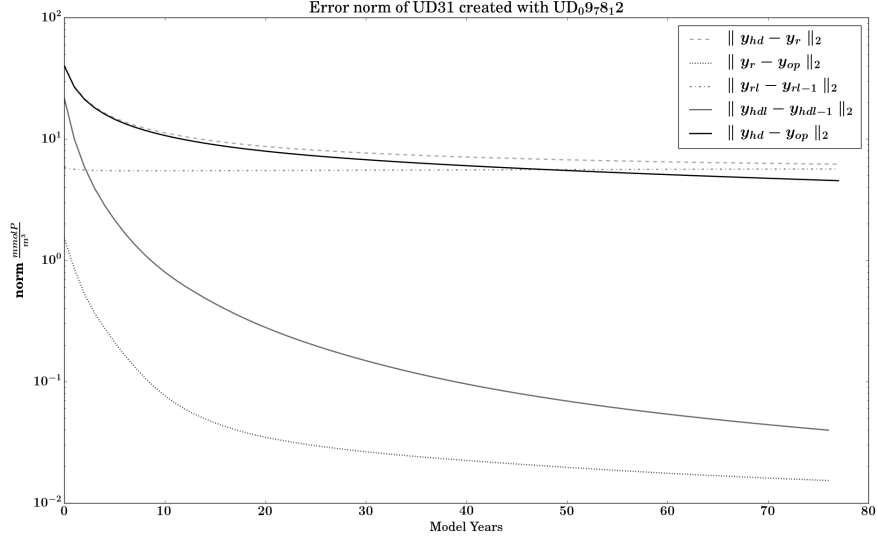


Abbildung 5: Vergleich des reduzierten Modelllaufs mit dem hochdimensionalen mit der Anfangsverteilung UD 31. Modellzeit 78 Jahre, gestrichelte Linie: Unterschied zwischen der hochdimensionalen und der reduzierten Lösung, graue gepunktete Linie: Fehler zur optimalen Verteilung des reduzierten Modelllaufs, gestrichelte und gepunktete Linie: Abstand zum Vorjahr, des reduzierten Modelllaufs, durchgezogene, graue Linie: Abstand zum Vorjahr des hochdimensionalen Modelllaufs, durchgezogene, schwarze Linie: Fehler zur optimalen Verteilung des hochdimensionalen Modelllaufs. Deutliche Abweichung sichtbar.

### 5.3 Experiment 3

Hier wurden die Snapshotsmatrix von Experiment 2 erweitert mit Lösungen von 25 verschiedenen Anfangsverteilungen, von denen allerdings nur die ersten 9 Jahre genommen wurden. Dies wurde gemacht, um eine Basis zu bekommen, die auch für unterschiedliche Anfangsverteilungen Ergebnisse liefert, die nah an den hochdimensionalen liegen. Auch hier ist zu erkennen, dass zum Ende die Singulärwerte deutlich abfallen (Abbildung 6).

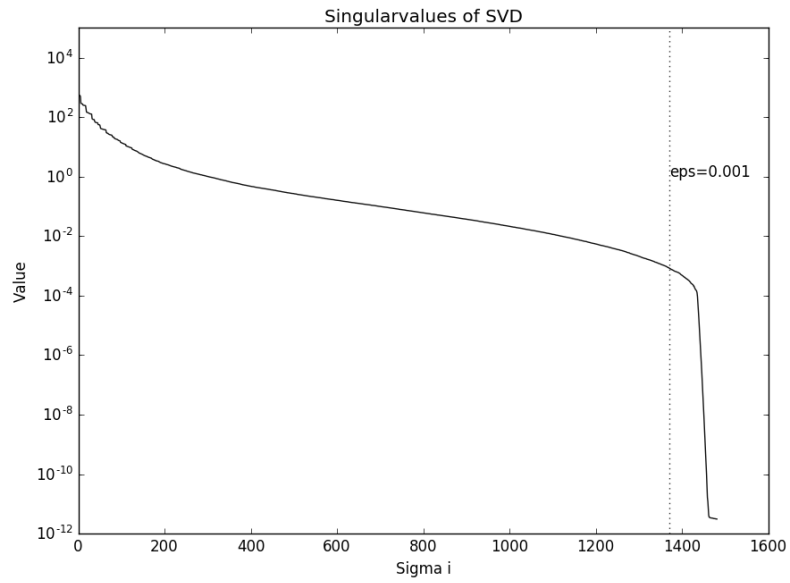


Abbildung 6: Singulärwerte von UD 09 78 + UD 25 50 9

Mit dieser Basis sind die Werte für eine Anfangsverteilung, die nicht in der Snapshotmatrix enthalten ist, näher an der hochdimensionalen (Abbildung 7).



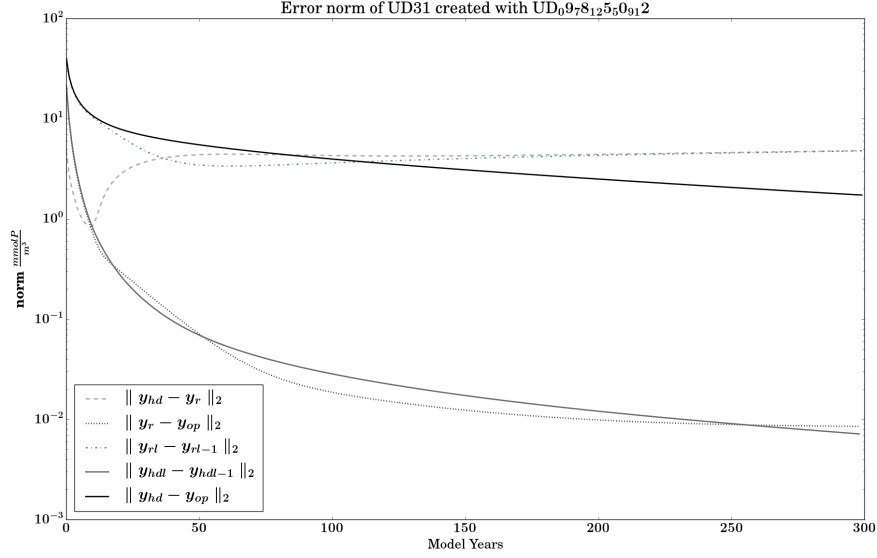


Abbildung 7: Vergleich des reduzierten Modelllaufs mit dem hochdimensionalen mit der Anfangsverteilung UD 31. Modellzeit 300 Jahre, gestrichelte Linie: Unterschied zwischen der hochdimensionalen und der reduzierten Lösung, graue gepunktete Linie: Fehler zur optimalen Verteilung des reduzierten Modelllaufs, gestrichelte und gepunktete Linie: Abstand zum Vorjahr, des reduzierten Modelllaufs, durchgezogene, graue Line: Abstand zum Vorjahr des hochdimensionalen Modelllaufs, durchgezogene, schwarze Line: Fehler zur optimalen Verteilung des hochdimensionalen Modelllaufs.

## 6 Komplexität und Konvergenz

Das Aufwändigste an dem POD-Verfahren ist das Generieren der Snapshots und das Berechnen der Singulärwertzerlegung. Für die Snapshots muss das hochdimensionale Modell ausgewertet werden, dies ist entsprechend der Komplexität des Modells teuer. Die Singulärwertzerlegung für  $A \in \mathbb{R}^{m \times n}$  ist in  $O(m^2n + mn^2 + n^3)$ , der Speicherbedarf für eine komplette Zerlegung ist bei den Daten von Metos3d leider sehr hoch, da eine Snapshotmatrix die Dimension  $52749 \times N$  hat und für die Zerlegung eine  $52749 \times 52749$  voll belegte Matrix  $U$  erstellt werden müsste. Also etwa 20GB bei Float64. Es werden aber nur die ersten  $k$  Spalten dieser Matrix  $U$  gebraucht, da nur die ersten  $k$  Singulärwerte ungleich 0 sind. Deshalb kann eine “Thin SVD” berechnet werden, bei der die Matrix  $U$  nur noch  $52749 \times k$  groß ist, auch die Komplexität liegt hier nur noch in  $O(m^2k)$ .

Die Berechnung der 12 Monatsmittelmatrizen ist in  $O(m^2k)$ , allerdings sind die Monatsmittelmatrizen des hochdimensionalen Modells dünn besetzt, also ist

diese Operation abhängig von der Anzahl  $p$  der Nicht-Null Einträge. Somit lässt sich diese Operation in  $O(pk)$  erledigen. All das sind allerdings Operationen, die vor der Auswertung des reduzierten Modells berechnet werden können.

Die Auswertung des reduzierten Modells ist abhängig von der Dimension  $k$  der Basis  $V$ . Es wird in jedem Zeitschritt eine Matrix-Vektor-Multiplikation der Dimension  $k$  durchgeführt, also  $k^2$  Operationen. Im Vergleich zum hochdimensionalen Modell, bei dem ein Zeitschritt in  $O(p)$  liegt, mit  $p$  der Anzahl von Nicht-Null Einträgen in der Systemmatrix, ist das eine Reduktion, wenn  $k < \sqrt{p}$ . In Metos3d ist  $p \approx 6 * 10^6$ , also sollte  $k < 2500$  sein. Dies ist bei allen Experimenten der Fall gewesen. Dort wurde meistens ein  $k$  gewählt, das kleiner war als 1000, damit ist also weniger als ein 1/6 der Rechenzeit des hochdimensionalen Modells notwendig gewesen.

Die Konvergenz gegen die hochdimensionale Lösung ist bis auf den Fehler  $\epsilon_{pod}$  der Basis gegeben (Abbildung 8), allerdings könnte es schwer sein diesen für sehr lange Simulationen niedrig genug zu halten.

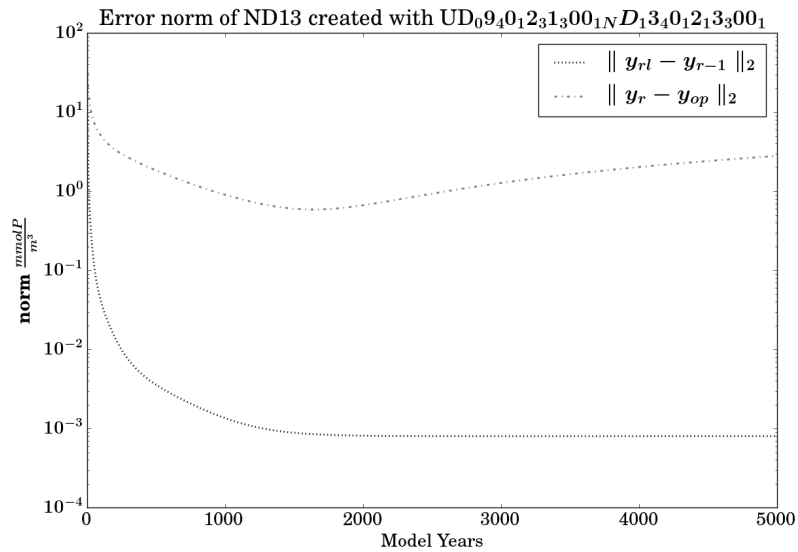


Abbildung 8: Konvergenz Untersuchung von reduzierten Modell, Modellzeit 5000 Jahre, Modell Fehler  $\epsilon_{pod} = 1e-3$

Das hochdimensionale Modell unterliegt einer monatliche Schwankung (Abbildung 9), die das reduzierte Modell nicht mehr hat, in wie weit das von Bedeutung ist muss noch untersucht werden.

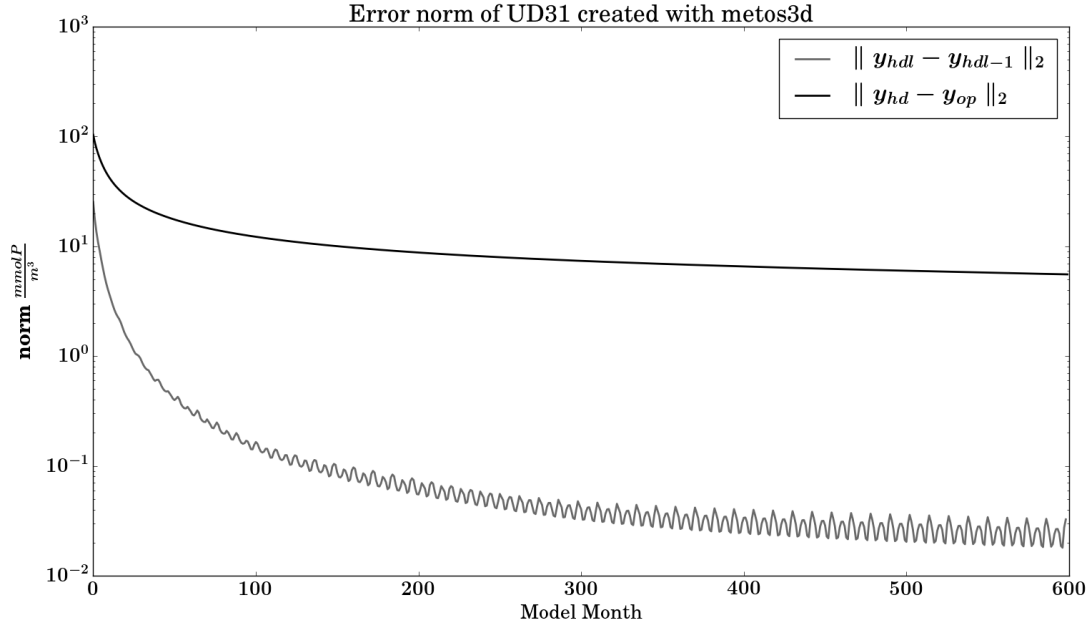


Abbildung 9: Monatliche Schwankung des hochdimensionalen Modell, nicht vorhanden beim reduzierten Modell

## 7 Code

Das POD-Verfahren wurde in Python umgesetzt, dabei wurde die numpy und scipy Bibliotheken für numerisches und wissenschaftliches Rechnen benutzt. Der Code kann unter [https://github.com/neeljp/pod\\_deim](https://github.com/neeljp/pod_deim) gefunden werden.

## 8 Fazit

Die Modellreduktion des linearen Anteils ist mit dem POD-Verfahren möglich und die Ergebnisse sind, bei entsprechend gewählter Basis, zufriedenstellend. Für die Auswahl der Snapshots empfiehlt es sich, Lösungen verschiedener Anfangsverteilungen zu nehmen, aber auch längere Trajektorien mit einzubeziehen. Das reduzierte Modell ist stabil, aber es konvergiert nicht unbedingt gegen die Lösung des hochdimensionalen Modells, da irgendwann die Basisgenauigkeit erreicht ist. Es könnte noch untersucht werden, ob die Matrizen des reduzierten Modells die Masse der Tracers verändern. Als nächstes könnte das DEIM-Verfahren für den

nicht linearen Teil getestet werden. Dazu müssten Snapshots von der nichtlinearen Funktion erstellt werden und auf diese das POD-Verfahren angewandt werden.

## Literatur

- [0] Dimitriu, Gabriel and Navon, Ionel M. and Ștefănescu, Răzvan “Application of POD-DEIM Approach for Dimension Reduction of a Diffusive Predator-Prey System with Allee Effect” *In: Large-Scale Scientific Computing: 9th International Conference, 2013. Revised Selected Papers*, 373–381
- [1] Alfio Quarteroni, Andrea Manzoni and Federico Negri “Reduced Basis Methods for Partial Differential Equations” *Springer International Publishing*, 296 Pages
- [2] D. B. Szyld “The Many Proofs of an Identity on the Norm of Oblique Projections” *Numerical Algorithms*, 42:309-323, 2006
- [4] S. Chaturantabut “Dimension Reduction for Unsteady Nonlinear Partial Differential Equations via Empirical Interpolation Methods” Master’s thesis, Rice University, 2008.
- [5] Piwonski, J. and Slawig, T. “Metos3D: a marine ecosystem toolkit for optimization and simulation in 3-D - Simulation Package” *Geoscientific Model Development Discussions*, Volume 8, 2015, Page 4401–4451